

# Platform One Big Bang AMA



# Big Bang - The Name of a Team and a Product

- Big Bang is a team within the Platform One team.
- The Big Bang team is responsible for developing the IaC/CaC that makes up the “Big Bang Product”, which is basically a Hardened Kubernetes Cluster defined by IaC/CaC and managed using GitOps, and that can be cATO'd because it follows the DoD DevSecOps reference design.



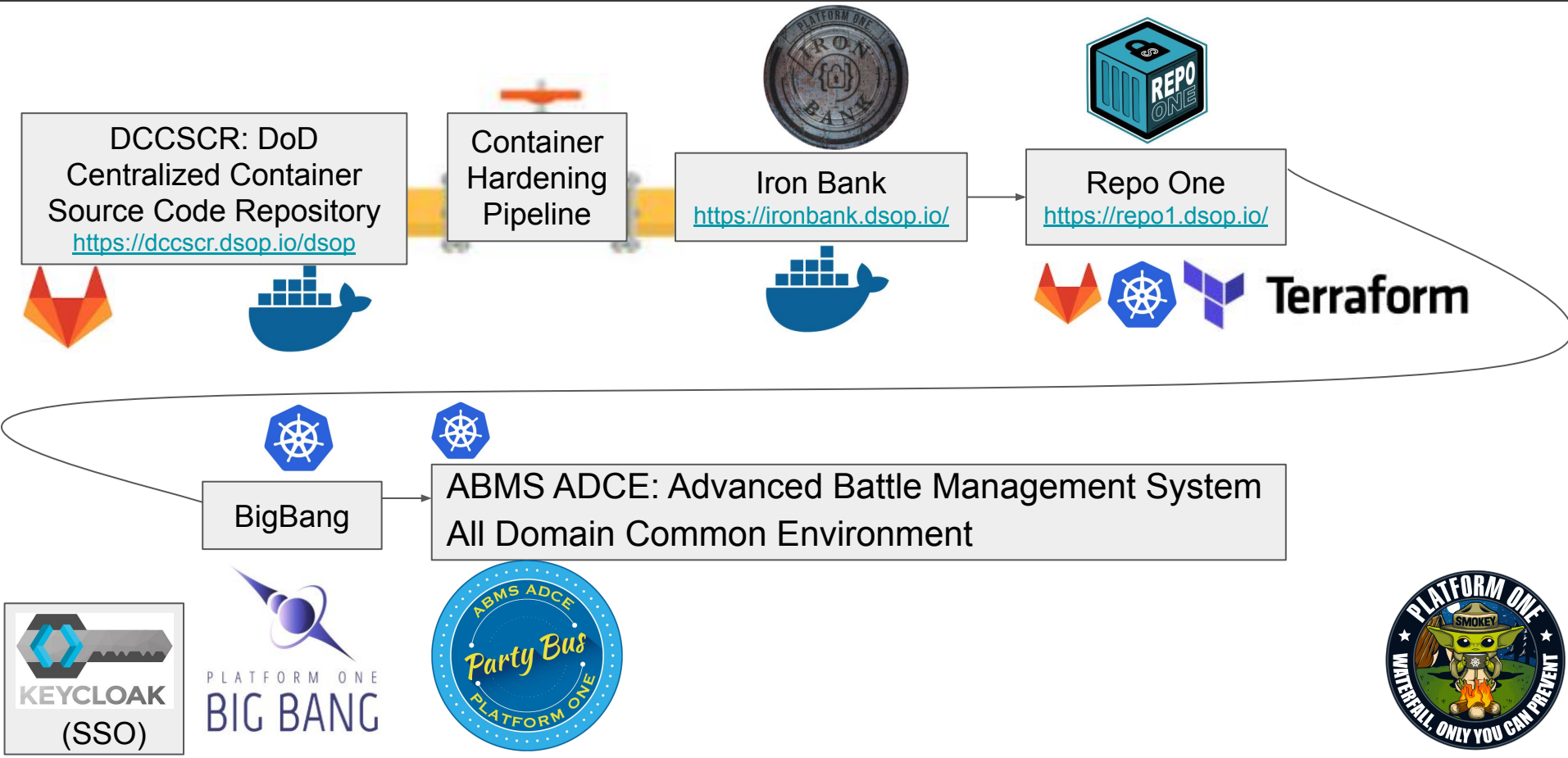
# Big Bang vs Party Bus

Party Bus/ABMS ADCE (Advanced Battle Management System All Domain Common Environment) is basically just Big Bang as a Service, they're both Hardened Kubernetes Clusters that can be cATO'd and used to host or develop applications on.

- Party Bus team manages the infrastructure provisioning and maintenance.
- Party Bus is a multi-tenant environment, and users of Party Bus get to inherit the Party Bus team's cATO.
- Big Bang is a better fit for users who need more control, want to allow their developers more access to operations tools in lower environments, or need air gapped deployments.



# How the P1 Services Fit Together



# Why Kubernetes? (1/2)

The answer ties back to Platform One's Mission: Streamlining the delivery of quality software to supports DoD Missions.

In the past it appeared that there were 2 choices in the software development world:

1. Slow and Safe (in terms of security and quality)
2. Fast and Risky (in terms of security and quality)
3. Fast and Safe (by leveraging DevSecOps and automating as much as possible, it's usually prohibitively expensive to unlock this 3rd option.)

Public Cloud Automation and PaaS's make Fast and Safe possible, but don't work for onprem or high classification levels.



# Why Kubernetes? (2/2)

Kubernetes Platform allows you to create DIY PaaS and automation that works anywhere.

So Both Public Cloud and Kubernetes can unlock the "Fast and Safe" software development option. But only Kubernetes can run OnPrem in the Cloud and at any classification level.



# What's so special about a Big Bang Kubernetes Cluster?

Kubernetes is like Agile and DevSecOps, where proper implementation is key:

It's easy enough to wrap your head around a big picture conceptual understanding of what it means and all the benefits you can get from it.

But when you actually try to implement any of these things for real, it's really easy to only get 2 out of 10 benefits, and come to the realization that mastering this and getting 10 out of 10 benefits simultaneously is really hard.

And What's Special about the Big Bang team is they do Kubernetes, Agile, and DevSecOps right. They develop the IaC using a DevSecOps pipeline.



# Big Bang Team's CI/CD Pipeline Deployment Workflow:

## Management VPC:

### Gitlab CI Runners hosted on Kubernetes



Runners run terraform  
commands based on  
branches.

## VPC Dynamically Spun up based on Branch "dev-<arbitrary name1>":

### Public Subnets (Multiple AZs)

Bastion



### Private Subnets (Multiple AZs)



Kubernetes Master 1

Kubernetes Master 2

Kubernetes Master 3

Kubernetes Worker 1

Kubernetes Worker 2

Kubernetes Worker 3

Utility VM (Private Docker Registry)



## VPC Dynamically Spun up based on Branch "dev-<arbitrary name2>":

### Public Subnets (Multiple AZs)

Bastion



### Private Subnets (Multiple AZs)



Kubernetes Master 1

Kubernetes Master 2

Kubernetes Master 3

Kubernetes Worker 1

Kubernetes Worker 2

Kubernetes Worker 3

Utility VM (Private Docker Registry)





# Big Bang Non-Pipeline or Air Gapped Deployment:

Mgmt /  
bootstrap VM  
(Git Repo)

## Public Subnets (Multiple AZs)

Bastion



## Private Subnets (Multiple AZs)



Utility VM (Private Docker Registry)



Kubernetes Master 1

Kubernetes Master 2

Kubernetes Master 3

Kubernetes Worker 1

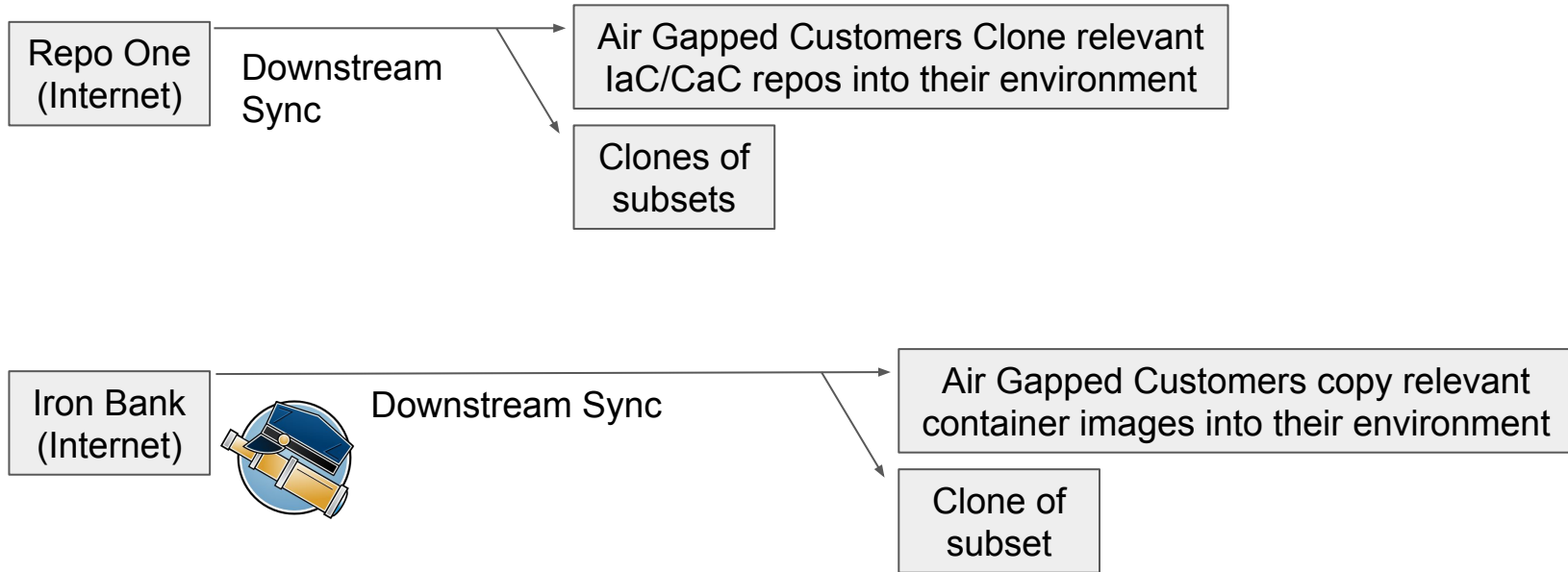
Kubernetes Worker 2

Kubernetes Worker 3



# Visualization of Repo One and Iron Bank

- Repo One: DoD GitLab Repo for IaC/CaC of Kubernetes Applications and Clusters
- Iron Bank: DoD Hardened Container Artifact Repo (Docker Image Registry)



# Visualization of Big Bang

- BigBang is a hardened Kubernetes Cluster defined by IaC/CaC and managed using GitOps.  
(That's meant for single tenancy)

## Customer A's Clusters

Internet Connected Environment  
Highest Automation and BB Support  
IaC/CaC hosted on Repo1



Dev

Mock Air Gapped Environment  
High Automation and BB Support  
IaC/CaC hosted on Repo1



Stage

Air Gapped Environment  
Semi Automation, less\* BB Support  
IaC/CaC hosted in Customer's Env



Prod

## Customer B's Clusters



Mission  
App 1  
(Prod)



Mission  
App 2  
(Prod)

## Customer C's Clusters



Dev / CI/CD  
Software  
Factory



Stage



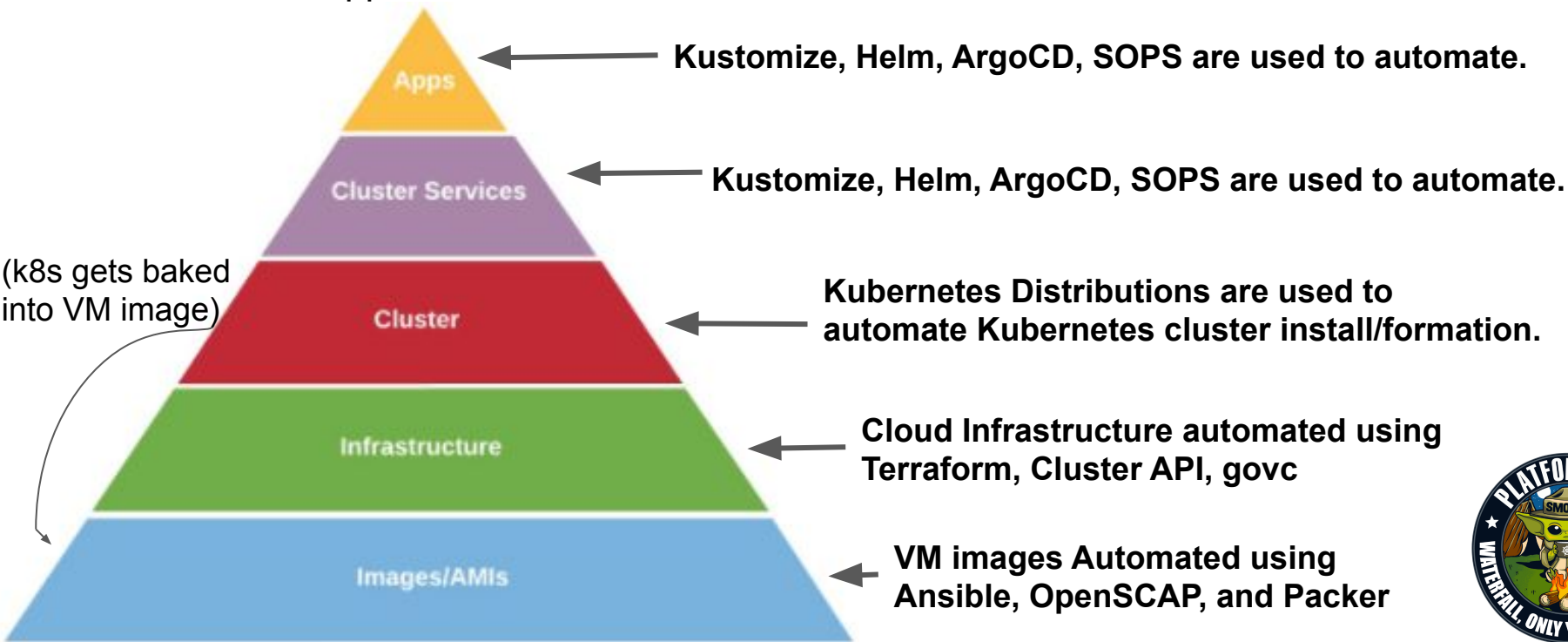
Prod  
(multiple mission apps)



# Big Bang Clusters follow Kubernetes and DevSecOps best practices:

Every Layer of the Kubernetes Cluster is defined as IaC/CaC and automated.

GitLabCI is used to "glue" the layers of automation into GitOps driven E2E automation of both infrastructure and applications



# Big Bang Clusters come in flavors:

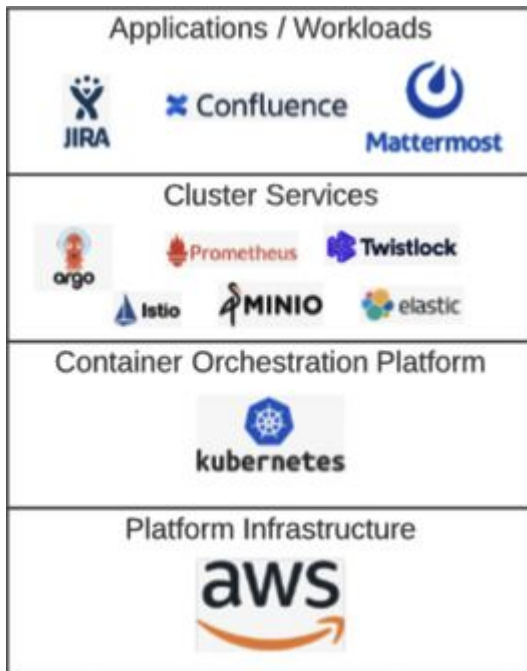
Top 2 Layers tend to have high reusability of IaC, due to Kubernetes Abstracting Complexity.  
Bottom 2 Layers tend to be very Platform Specific



- **Layer 4** – Applications consumed by the users of the platform.
- **Layer 3** – Set of services to manage and support the platform applications.
- **Layer 2** – Kubernetes distribution for automating computer application deployment, scaling, and management.
- **Layer 1** – Cloud provider or physical hardware for the compute infrastructure.



# Big Bang Clusters come in flavors:



## Customer's Choice of Applications:

Can run a single mission app, multiple mission apps, a Software Factory or all of the above.

## Applications to be Compliant with DevSecOps Reference Architecture:

Must\* run: ArgoCD, Istio, ElasticSearch, Prometheus, Twistlock

## Kubernetes Distribution Installation Options:

"archosaur" (kubeadm), BYO\*

Konvoy (Cloud Agnostic Managed Kubernetes Platform)

## Cloud Infra Options:

AWS (Stable), VMware (WIP Preview), Azure (Coming Soon)



# What applications are in a BigBang Cluster?

## **Platform level applications:**

**GitOps:** ArgoCD

**Monitoring/Logging:** ElasticSearch, Prometheus, Istio

**Service Mesh:** Istio

**Runtime Security:** Istio, Twistlock, Open Policy Agent

## **Optional Software Factory Applications:**

**CICD tooling:** Anchore and Twistlock (Vulnerability Scanning), GitLab, ArgoCD, Minio (Self Hosted S3), SonarQube (Code Quality), Jaeger (Application Performance Monitoring).

**Developer Collaboration tooling:** Keycloak (Federated Single Sign On), Confluence (documentation management), Jira (task management), MatterMost (Chat).

## **Other Optional Applications:**

<https://repo1.dsop.io/platform-one/apps>



# Big Bang AMA

Questions?

